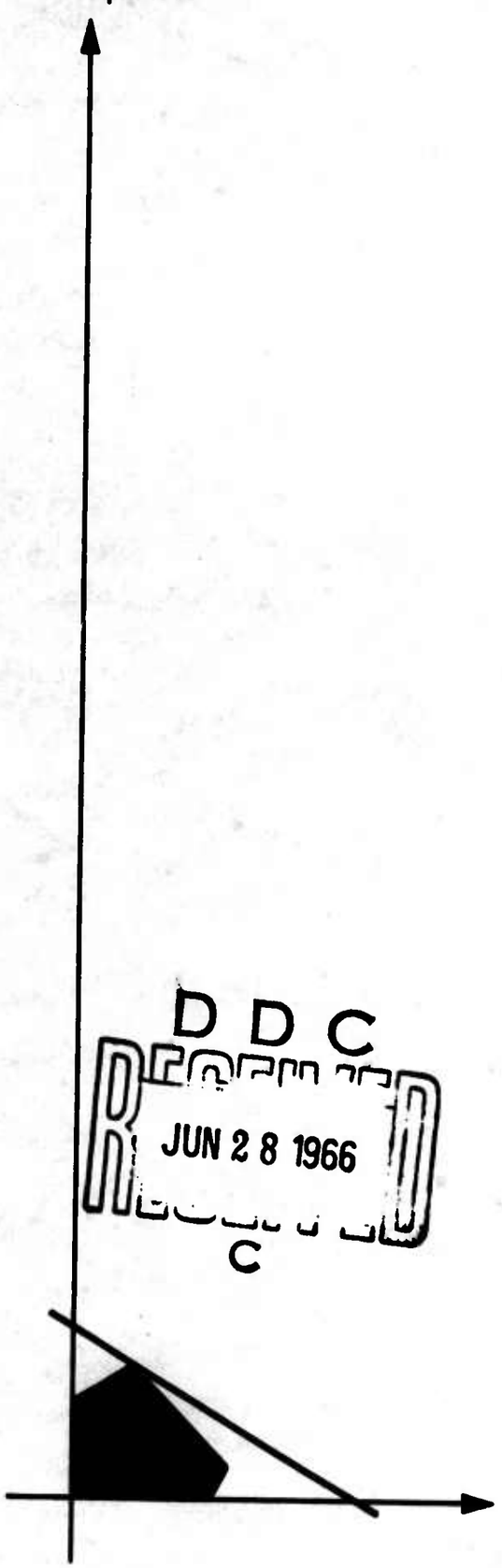
ORC 66-13
April 1966

NOTES ON OPERATIONS RESEARCH--5



OPERATIONS RESEARCH CENTER

COLLEGE OF ENGINEERING

UNIVERSITY OF CALIFORNIA - BERKELEY

NOTES ON OPERATIONS RESEARCH -- 5

Operations Research Center
University of California, Berkeley

April 1966 ORC 66-13

This research has been partially supported by the Office of Naval Research under Contract Nonr-222(83), the National Science Foundation under Grant GP-4593, and the U.S. Army Research Office-Durham, Contract DA-31-124-ARO-D-331 with the University of California.

CONTENTS

NOTES ON OPERATIONS RESEARCH -- 5

ARTICLES

Remarks about Random Sampling from Beta and Gamma Distributions
Bennett Fox 1

Hanson's Dual Program
R. N. Kaul 7

A Short Proof of Hu's Assembly Line Theorem
Lars-Chr. Lorentzen 12

Proof of a Theorem from the Theory of Complementary Solutions of Linear Systems
Lars-Chr. Lorentzen 17

Networks and Assembly Line Balancing
Shishir Kumar Mukherjee 21

Summary of Klee's Results on Bounding the Number of Simplex Cycles
Bennett Fox 28

REVIEW

Cycling in the Transportation Problem -- by B.J. Gassner
Bennett Fox 31

RESEARCH PROBLEMS
George B. Dantzig 32

ABSTRACTS

Selected Thesis Abstracts in Operations Research 36

REMARKS ABOUT RANDOM SAMPLING FROM BETA AND GAMMA DISTRIBUTIONS

by Bennett Fox

Methods for generating random numbers from beta and gamma distributions exist (see the appendix), but the time required to generate them is an order of magnitude larger than that required when the parameters are integers.

From a gamma distribution with integral shape parameter, random numbers can readily be generated by well-known methods. For the beta distribution with integral parameters (a,b), an efficient method was pointed out by the author in [1] and independently noted later in [2]. The technique is to take the ath smallest of $a + b - 1$ uniform random numbers.

Rather than generate random numbers from a gamma distribution with non-integral shape parameter, we suggest that one look to the Weibull family, from which one can usually get a equally good fit to data. A great many observations are required to discriminate between the Weibull and gamma families. Relative to known methods (see, e.g., the appendix) for generating random numbers from a gamma distribution with nonintegral shape parameter, a random number w from the Weibull distribution with c.d.f. $1 - e^{-\lambda x^k}$ can quickly be obtained by setting $w = (-\frac{1}{\lambda} \log u)^{1/k}$, where u is a uniform random number.

Now we dispose of the case of a beta distribution with nonintegral parameters. There are two natural approximate methods. One is to use a beta distribution with integral parameters close (with respect to some norm) to the original.

As an example of integral approximation, consider a Bernoulli process (e.g., coin flips) with parameter p assumed to have a beta prior distribution $\beta(a,b)$. More explicitly, suppose that p has a fixed (but unknown) value p_0 and that $\beta(a,b)$ measures our uncertainty about its location. One

way to simulate the process (not necessarily the most efficient) is to generate a random from $\beta(a,b)$, perform a Bernoulli trial with this random number as parameter, and repeat. Suppose that we wish to choose appropriate values for a and b from empirical data. With k successes (e.g., heads in coin flips) in n trials, the method of matching moments yields

$$\frac{\hat{a}}{\hat{a} + \hat{b}} = \frac{k}{n}$$

$$\frac{\hat{a}\,\hat{b}}{(\hat{a} + \hat{b})^2(\hat{a} + \hat{b} + 1)} = \frac{(k/n)\,[1 - (k/n)]}{n}$$

If n is large, we have the immediate (near) integral solution.

$$\hat{a} \simeq k$$

$$\hat{b} \simeq n-k ,$$

which coincide with the parameters of the posterior distribution starting with a uniform prior distribution.

The second method is based on a step function approximation to the beta density function with, say, n steps. Let the height and width of the i-th step be h_i and w_i, respectively. Assume that $\sum_1^n h_i w_i = \sum_1^n w_i = 1$. Generate uniform random numbers u and u', and suppose that $\sum_1^{k-1} h_i w_i \leq u < \sum_1^{k} h_i w_i$. Then $u'w_k + \sum_1^{k-1} w_i$ is a random number from the approximating distribution. A similar technique works for trapezoidal approximation.

A case where sampling from an approximating distribution might be appropriate is the simulation of project networks (see, e.g., [3]) where the job durations are often assumed to have beta distributions, except possibly for a change of scale and translation. Presumably, these distributions represent the

a priori beliefs of the planner, who would seldom be willing to claim that he was sure that a job duration had *exactly* a particular beta distribution. Hence, it is reasonable to sample from an approximating distribution, especially when considerable computer time can be saved by doing so. As was pointed out in [3], it may be appropriate to use a triangular distribution.

APPENDIX

Exact schemes for sampling from beta and gamma distributions with arbitrary parameters have been proposed [2]. For convenient reference, we give them here. The proofs are straightforward; for details, see [2].

The sampling algorithm for the beta distribution $\beta(a,b)$ is

1. $i = 1$.
2. Generate a uniform random number u_i and set $x = u_i^{1/a}$.
3. Generate a uniform random number u_{i+1} and set $y = u_{i+1}^{1/b}$.
4. If $x + y \leq 1$, go to 6; otherwise, go to 5.
5. $i \to i + 2$; go to 2.
6. Done. $x/(x + y)$ is a random number from $\beta(a,b)$.

An (incomplete) measure of efficiency is $1/E$, where E is the expected number of uniform random numbers generated. This measure does not take account of all the calculations required. For the above algorithm,

$$E = \frac{2(a + b)}{a\, b\, B(a,b)} . \qquad \text{When } a = b = 3 , \quad E = 40 .$$

If in Step 6 $x/(x + y)$ is replaced by x, we get a random number from $\beta(a, b + 1)$. Hence for $b > 1$ this gives a slightly faster way of getting a random number from $\beta(a,b)$.

For the gamma distribution with nonintegral shape parameter c and scale parameter λ, the algorithm is

1. Set $z = -\log \prod_1^{[c]} u_i$, the u_i are uniform random numbers.
2. Generate a random number w from $\beta(c - [c], 1 - c + [c])$, using the preceding algorithm.

3. Generate a uniform random number u and set $y = -\log u$.

4. $(z + wy)/\lambda$ is the desired random number.

Letting $k = c - [c]$, we have

$$E = [c] + 1 + \frac{2}{k(1 - k)\, B(k, 1 - k)}$$

$$= [c] + 1 + \frac{2 \sin \pi k}{\pi k(1 - k)}$$

$$\simeq [c] + 3 .$$

REFERENCES

[1] Fox, B.L., 1963, "Generation of random samples from the beta and F distributions," *Technometrics*, 5, 269-270.

[2] Jöhnk, M.D., 1964, "Erzeugung von betaverteilten und gammaverteilten Zufallszahlen," *Metrika*, 8, 5-15.

[3] Van Slyke, R.M., 1963, "Monte Carlo methods and the PERT problem," *Operations Research*, 11, 839-860.

HANSON'S DUAL PROGRAM

by R.N. Kaul

1. *Introduction*

In [1] Hanson studies a pair of non-linear Problems I and II such that existence of optimal solution to one of them implies the existence of optimal solution to the other and the two extrema are equal. While the result that the existence of an optimal solution to Problem I insuring an optimal solution to Problem II is straightforward and has also been proved by Wolfe [2]; the converse problem, namely, the existence of optimal solution to Problem II implying an optimal solution for Problem I makes use of some assumptions. The aim of this note is to give an auxiliary condition similar to the one considered in [3] which ensures the existence of optimal solution to Problem I as well as the equality of the objective functions once an optimal solution to Problem I exists. We, of course, assume that the constraint qualifications of Kuhn and Tucker [4] are satisfied.

2. *Dual Programs*

Problem I: Minimize $\Phi(x)$

Subject to $f(x) \geq 0$ --- (1)

$x \geq 0$ --- (2)

Problem II: Maximize $G(x,y) \equiv F(x,y) - x^T \nabla_x F(x,y)$

Subject to $\nabla_x F(x,y) \geq 0$ --- (3)

$y \geq 0$ --- (4)

where $F(x,y) = \Phi(x) - y^T f(x)$

Here $\Phi(x)$ is a convex differentiable function of the vector x, $f(x)$ is a concave differentiable function of the n-dimensional vector x and $\nabla_x = (\partial/\partial x_1, \ldots, \partial/\partial x_n)^T$.

Let x_o be the optimal solution for Problem I. Then according to the conditions of Kuhn and Tucker [4], there exists a vector y_o such that

$$y_o \geq 0 \tag{5}$$

$$[\nabla_x F(x,y)]_{x_o,y_o} \geq 0 \tag{6}$$

$$y_o^T f(x_o) = 0 \tag{7}$$

$$x_o^T [\nabla_x F(x,y)]_{x_o,y_o} = 0 \tag{8}$$

For any feasible solution (x,y) of Problem II, we have using (7) and (8)

$$G(x_o,y_o) - G(x,y) =$$

$$\Phi(x_o) - \Phi(x) + y^T f(x) + x^T \nabla_x \Phi(x) - x^T \nabla_x (y^T f(x)) \; . \tag{9}$$

Also

$$\Phi(x_o) - \Phi(x) \geq (x_o - x)^T \nabla_x \Phi(x) \tag{10}$$

because $\Phi(x)$ is convex

and

$$y^T f(x) \geq y^T [f(x_o) - (x_o - x)^T \nabla_x f(x)] \tag{11}$$

because $f(x)$ is concave

Therefore making use of the above inequalities together with (1), (2), (3), (4) and (8) the relation (9) yields

$$G(x_o,y_o) - G(x,y) \geq 0$$

which establishes that (x_o, y_o) is an optimal solution to Problem II.

The above result has already been proved in [1] and [2].

The main result that shall be proved is the following:

If $(\bar{x}, \bar{y})$ is an optimal solution to Problem II and if the following auxiliary condition is satisfied, then there exists an optimal solution to Problem I and the optimum of the objectives of the two problems are equal.

Condition A: The matrix $\nabla_{xx} F(x,y)$ is non-singular at $(\bar{x}, \bar{y})$.

Assume now that we have an optimal solution $(\bar{x}, \bar{y})$ to Problem II. Applying again the conditions of Kuhn Tucker the following holds.

There exists a vector $\bar{\lambda} \geq 0$ such that

$$\bar{\lambda}^T \nabla_{xx} F(x,y) + \nabla_x G(x,y) = 0$$

$$\bar{\lambda}^T \nabla_{yx} F(x,y) + \nabla_y G(x,y) \leq 0$$

$$\bar{\lambda}^T \nabla_x F(x,y) = 0$$

$$[\lambda^{-T} \nabla_{yx} F(x,y) + \nabla_y G(x,y)] \bar{y} = 0$$

All the derivatives in the above relations are evaluated at $(\bar{x}, \bar{y})$. These reduce to

$$(\bar{\lambda}^T - \bar{x}^T) \nabla_{xx} F(x,y) = 0 \tag{12}$$

$$(\bar{x}^T - \bar{\lambda}^T) \nabla_x f(x) - f(x) \leq 0 \tag{13}$$

$$\bar{\lambda}^T [\nabla_x F(x,y)] = 0 \tag{14}$$

$$[(\bar{\lambda}^T - \bar{x}^T) \nabla_x f(x) - f(\bar{x})] \bar{y} = 0 \tag{15}$$

From condition A and (12) it follows that

$$\bar{\lambda}^T = \bar{x}^T$$

Thus relations (13), (14), (15) imply respectively (1), (8) and (7). We know by assumption that relations (5) and (6) are true. This shows that the necessary conditions for the existence of optimal solution to Problem I are satisfied.

But since $\Phi(x)$ is convex and $f(x)$ concave, these necessary conditions are also sufficient [5]. Hence the result.

REFERENCES

[1] M.A. Hanson, "A Duality Theorem in Non-linear Programming with Non-linear Constraints," Austral. Jour. Statist., Vol. 3, 1961, pp. 64-72.

[2] P. Wolfe, "A Duality Theorem for Non-Linear Programming," Q. App. Math., Vol. 19, 1961, pp. 239-244.

[3] Pierre Huard, "Dual Programs. Recent Advances in Mathematical Programming," edited by P. Wolfe and R.L. Graves, McGraw-Hill, 1963.

[4] H.W. Kuhn and A.W. Tucker, "Non-linear Programming," Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, J. Neyman, editor; Berkeley and Los Angeles, California, University of California Press, 1951, pp. 481-492.

[5] G. Hadley, "Non-linear and Dynamic Programming," Reading Mass: Addison-Wesley, 1964.

A SHORT PROOF OF HU'S ASSEMBLY LINE THEOREM

by Lars-Chr. Lorentzen

1 Introduction:

T.C. Hu has in [1] studied the situation where n jobs, taking the same amount of time have to be done by m men. The jobs are partially ordered where job A dominates job B means that job A has to be done before job B. He gives a procedure for finding an assignment of men to jobs which minimizes the time requ red to complete all the jobs in the case where the partially ordered set of jobs forms a forest with every tree having a unique minimal job. The purpose of this note is to give a shorter proof of the fact that the resulting assignment is optimal.

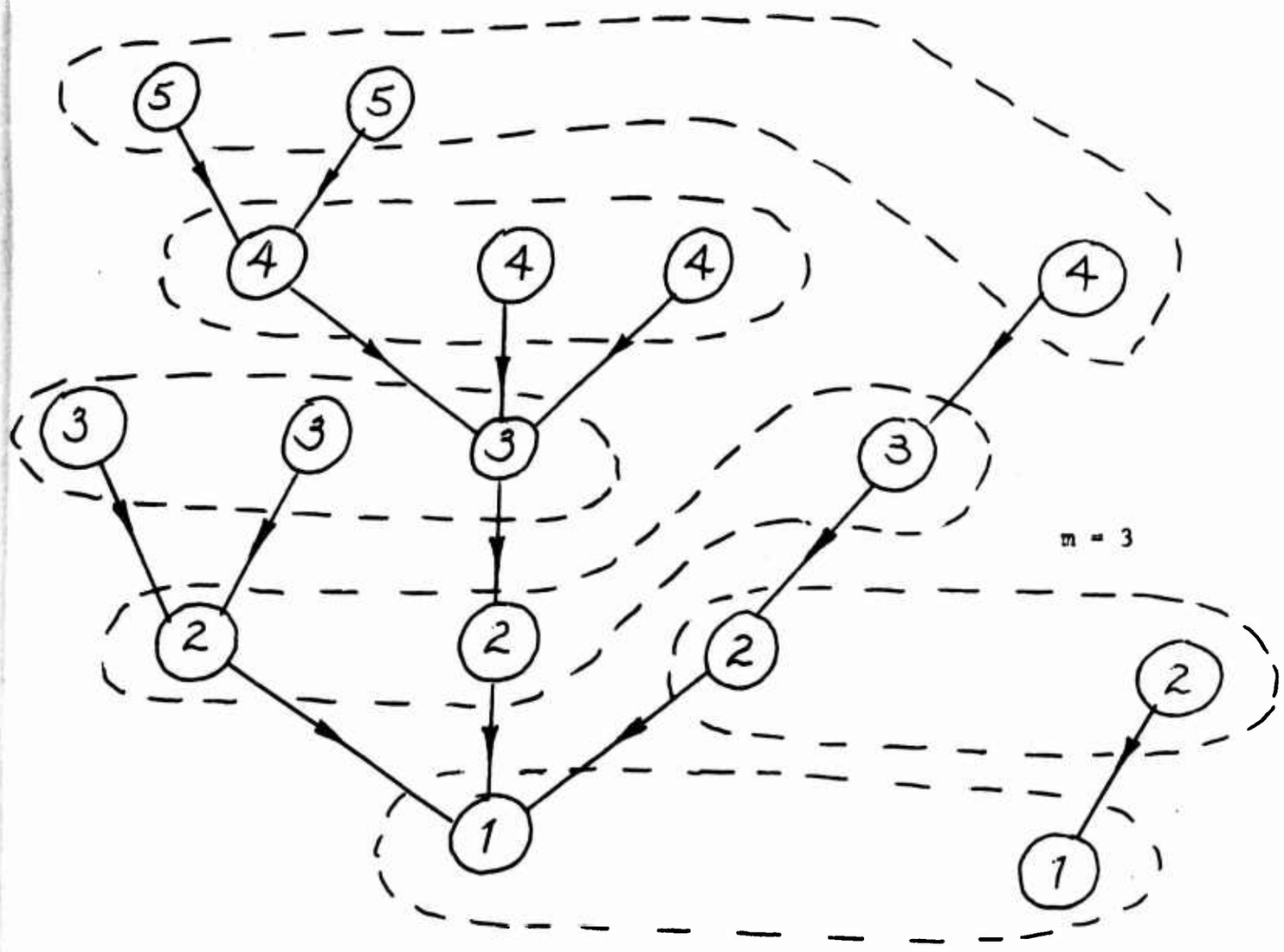
2 Statement of the Procedure:

First we label all the jobs in the forest in steps. Starting with i=1 we, in the i-th step, give all the minimal jobs in the subforest of unlabeled jobs the label i .

Then we assign men to jobs:

Starting with i = 1 we in the i-th time period assign maximal jobs in the subforest of incompleted jobs to as many of our m men as possible. If there are more than m maximal jobs we give preference to the jobs with highest labels.

Example:



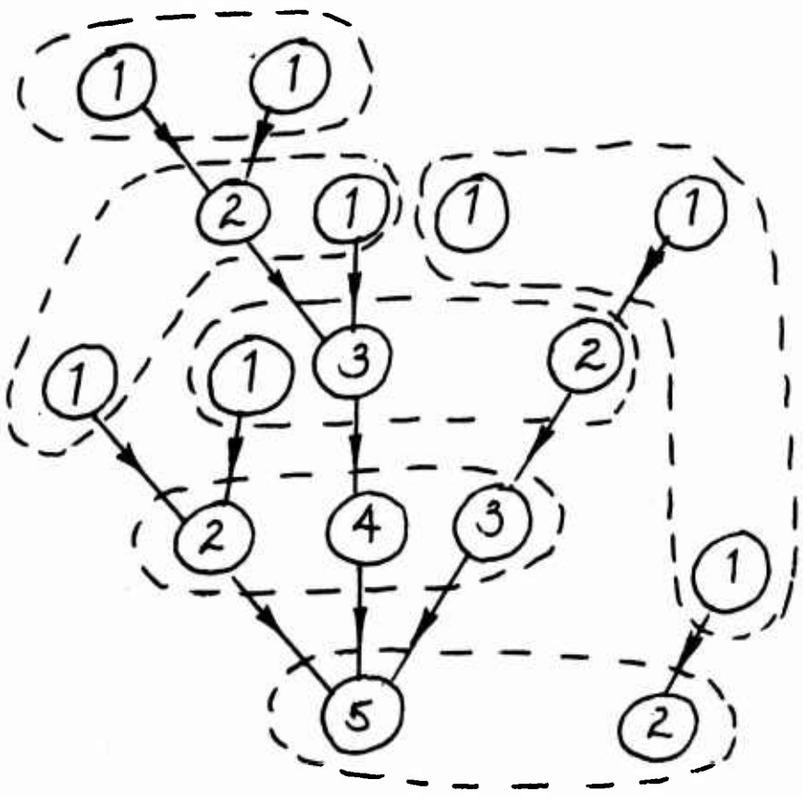
Observe that the procedure need not specify the assignment uniquely.

3 Reversed Procedure:

We shall now describe a reversed procedure. First we label every job with the "distance" from the "farthest" maximal job which dominates it. Then we assign men to jobs backwards in time:

Starting with $i = 1$ we in the i-th period from finishing time assign minimal jobs in the subforest of uncompleted jobs to as many of our m men as possible. If there are more than m minimal jobs we give preference to the jobs with highest labels.

Our example:



4 Proof of the fact that the two procedures lead to optimal assignments:

First we consider the reversed procedure:

Suppose the result is proved for any forest of $n - 1$ or less jobs, and let a forest of n jobs be given. If the forest has $\leq m$ minimal jobs, an optimal last step is obviously to complete all the minimal jobs and we are back to the case with fewer than n jobs. Let us assume that the forest has $> m$ minimal jobs. Let a possible set of jobs to be done in the last time period following the reversed procedure be $\alpha_1 \ldots \alpha_m$ where the α's represent the labels, but assume this is not an optimal way of finishing. Let it be optimal to complete the jobs $\beta_1 \ldots \beta_r \; \alpha_1 \ldots \alpha_{m-r}$ in the last time period. We now know that it is optimal to continue according to the procedure because of the induction hypothesis. In the next to last period we therefore need not complete any job dominating $\beta_1 \ldots \beta_r$ because if $\alpha_i > 1$ for $i = 1,\ldots,n$ we can find m minimal jobs

$\alpha_1 - 1 \ldots \alpha_{m-r} - 1 \; \alpha_{m-r+1} \ldots \alpha_m$ with labels $\geq \beta_i - 1$, $i = 1,\ldots,r$,

and if some $\alpha_i = 1$ there is no job dominating $\beta_1 \ldots \beta_r$. Assume an optimal next to last step is to complete the jobs

$$\alpha_1 - 1 \ldots \alpha_s - 1 \quad \alpha_{m-r+1} \ldots \alpha_m \quad \underbrace{\gamma_1 \ldots \gamma_t}_{\text{minimal jobs in the original forest}}$$

where $s + t \leq r$.

But these two steps are equivalent to completing the jobs $\alpha_1 \ldots \alpha_m$ in the last time period and the jobs

$$\beta_1 \ldots \beta_r \quad \alpha_1 - 1 \ldots \alpha_s - 1 \quad \gamma_1 \ldots \gamma_t$$

in the next to last time period.

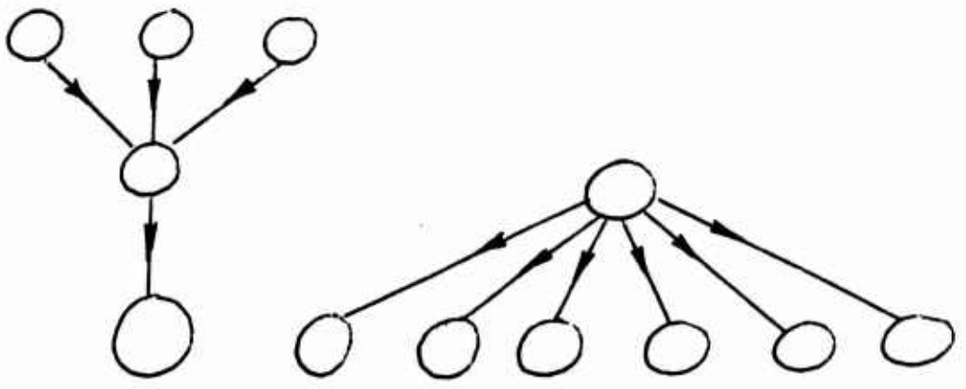
This completes the proof for the reversed procedure.

We now turn to the original procedure. Assume that the result is proved for a forest of $n - 1$ or less jobs and we are given a forest of n jobs. Suppose it would be compatible with the procedure to start doing jobs $\alpha_1 \ldots \alpha_m$. It is easily seen that if we perform the reversed procedure giving preference to jobs with lowest "forward" labels when this occurs, we can secure that in the first time period a subset of the jobs $\alpha_1 \ldots \alpha_m$ is done. The result therefore follows by induction.

5 The problem of generalization:

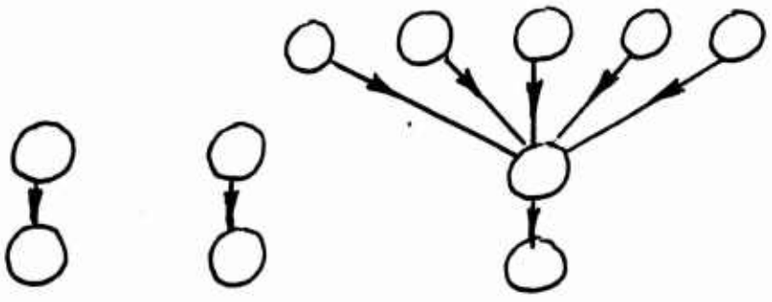
a) The following example shows that the method does not work if the trees do not have unique minimal jobs:

$m = 3$



b) It is easily seen that our proof is valid if the number of available men is increasing with time. The following example shows however that when we have the more realistic situation where the number of men m(t) is decreasing in time the procedure does not work:

t	1	2	3	4	5	...
m(t)	5	4	1	1	1	...



6 Reference: T.C. Hu, "Parallel Sequencing and Assembly Line Problems," *Operations Research* 9, 6, 1962.

PROOF OF A THEOREM FROM THE THEORY OF COMPLEMENTARY SOLUTIONS OF LINEAR SYSTEMS

by Lars-Chr. Lorentzen

Given the linear system

$$w = Mz + q$$

and any complementary canonical equivalent

$$\bar{w} = \overline{Mz} + \bar{q} \qquad \text{(See [1])}$$

The objective of this note is to give a proof of the following theorem stated in [1] p. 7:

Theorem: If M has nonnegative (positive) principal determinants, so has $\bar{M}$.

Lemma 1: If M has nonnegative (positive) principal determinants, then $M^*(\xi) = \xi M + (1 - \xi)I$ has positive principal determinants for $0 \leq \xi < 1$ ($0 \leq \xi \leq 1$ in the positive case).

Proof: Let N be a principal minor of M of order p. The by expansion we see that

$$\det N^*(\xi) = \det \xi N + (1 - \xi)^p + \sum_{i=1}^{p-1} (1 - \xi)^i \sum_j \det \xi P_j^{(p-i)} > 0$$

where $P_j^{(p-i)}$ is the j-th principal minor of the $(p - i)$-th order.

Lemma 2: If M has nonvanishing principal determinants, so has $\bar{M}$.

Proof: Without loss of generality we may assume that $\bar{M}$ results from M by pivoting on m_{11} ($\neq 0$ by hypothesis) and interchanging z_1 and w_1. We distinguish between two types of principal minors: those which contain m_{11} and those which do not. It suffices to consider $M_1 = [m_{ij}]$ $i,j = 1,\ldots,p$ and $M_2 = [m_{ij}]$ $i,j = 2,\ldots,p$. The rank of M_1 does not change by the interchange because $[1, 0, \ldots, 0]^T$ is independent of $M_1^2,\ldots,M_1^p$ where M_j^i is the i-th column of M_j. (Indeed if we had dependence, $M_2^2,\ldots, M_2^p$ would have been dependent, which is a contradiction since M_2 is nonsingular.)

Neither does the pivot change the rank of M_1 . As far as $\overline{M}_2$ is concerned we see at once that $\det \overline{M}_2 = \frac{1}{m_{11}} \det M_1 \neq 0$ as the interchange of w_1 and z_1 does not affect this minor.

Lemma 3: The principal determinants of $\overline{M^*(\xi)}$ are continuous functions of ξ for $0 \leq \xi \leq 1$.

Proof: The transition from $M^*(\xi)$ to $\overline{M^*(\xi)}$ corresponds to premultiplying $w = Mz + q$ by a block pivot matrix of the form

$$\begin{bmatrix} -A^{-1}(\xi) & , & 0 \\ C(\xi)A^{-1}(\xi) & , & -I \end{bmatrix}$$

where we have written

$$M^*(\xi) \text{ as } \begin{bmatrix} A(\xi) & , & B(\xi) \\ C(\xi) & , & D(\xi) \end{bmatrix}$$

Thus the only denominator in the expression for $M^*(\xi)$ is

$$\det A(\xi) = \det[\xi A + (1-\xi)I] \geq \det \xi A + (1-\xi)^a \geq 1/2 \min (1, \det A)$$

The proof of the theorem now goes like this:

Let $N^*(\xi)$ be a principal minor in $M^*(\xi)$

Then $\det \overline{N^*(\xi)} \neq 0$ for $0 \leq \xi < 1$ ($0 \leq \xi \leq 1$ in the positive case).

$N^*(\xi)$ is continuous for $0 \leq \xi \leq 1$, and $N^*(0) = 1$.

Dantzig has proved the following which can replace lemma 2:

Lemma 2': If M has positive principal determinants, so has $\overline{M}$.

Proof: As earlier we assume that $\overline{M}$ results from M by pivoting on m_{11} and then interchanging z_1 and w_1 , and we consider M_1 and M_2 .

We have $\bar{M}_1 = \begin{bmatrix} \frac{1}{m_{11}} & -\frac{m_{12}}{m_{11}} & \cdots \\ \frac{m_{21}}{m_{11}} & m_{22} - m_{21}\frac{m_{12}}{m_{11}} & \\ \cdots & & \end{bmatrix}$

Multiplying row 1 by m_{i1} and subtracting from row i $(i = 2,\ldots,p)$ makes it clear that $\det \bar{M}_1 > 0$.

Furthermore we have

$$\bar{M}_2 = \begin{bmatrix} m_{22} - m_{21}\frac{m_{12}}{m_{11}} & \cdots \\ \cdots & \end{bmatrix}$$

$$\text{so } \det \bar{M}_2 = \begin{vmatrix} 1 & \frac{m_{12}}{m_{11}} & \cdots \\ 0 & m_{22} - m_{21}\frac{m_{12}}{m_{11}} & \\ \cdots & & \end{vmatrix}$$

Multiplying row 1 by m_{i1} and adding to row i makes it clear that $\det \bar{M}_2 > 0$.

The type of argument used in the proof of lemmas 1 and 3 can also be applied to proving the following

Theorem: If M is positive semidefinite (definite), then $\det M \geq 0$ (> 0) .

Proof: $\xi M + (1 - \xi)I$ is positive definite for $0 \leq \xi < 1$ $(0 \leq \xi \leq 1$ in the definite case) and has therefore nonvanishing determinant. Since it is a continuous function of ξ in [0,1] with value 1 for $\xi = 0$ the theorem follows.

REFERENCE

[1] G.B. Dantzig and R.W. Cottle, "Positive (Semi-) Definite Matrices and Mathematical Programming," ORC 63-18 (RR), 13 May 1963.

NETWORKS AND ASSEMBLY LINE BALANCING

by
Shishir Kumar Mukherjee

The assembly line balancing problem belongs to the class of combinational optimization problems theoretically solvable by complete enumeration, but which has defied solution by efficient algorithms. In a line balancing problem, a finite set of work elements are given, each having a fixed processing time. The set is partially ordered and a set of precedence relations are given which specify the permissible ordering of the elements. The main problem of line balancing is to assign the elements to an ordered sequence of stations such that the precedence relations are not violated and the sum of the delay time at each station is minimized.

Formally we follow Gutjahr and Nemhauser [2]* to define the problem as follows: Given a finite set B, a partial order p defined on B, a positive real valued function t defined on B, and a number c, find a collection of subsets of B, $(A_1,\ldots,A_n)$, satisfying the following five conditions:

(1) $\bigcup_{i=1}^{n} A_i = B$

(2) $A_i \cap A_j = \emptyset$ (the empty set), $i \neq j$

(3) $t(A_i) = \sum_{x \in A_i} t(x) \leq c, \quad i = 1,\ldots,n$

(4) If xpy (x precedes y) and $x \in A_i$, $y \in A_j$, then $i \leq j$

(5) $\sum_{i=1}^{n} (c - t(A_i))$ is minimized.

The set B is the set of the work elements (x,y,...) and the partial order p is the set of precedence relations; t represents the required time and c is the cycle time. Sets A_i satisfying 1-4 are feasible assignments to work stations which are usually too many, and those satisfying all five conditions are an assignment of elements to the n-stations which minimize

*Numbers in parentheses correspond to references at the end of this paper.

the total delay time for any n.

The literature on assembly line balancing contains a number of methods and a good summary of some of the earlier methods may be found in Kilbridge and Wester [7]. Salveson [12] authored the first published paper on assembly line balancing in 1955. Jackson [6] presents an algorithm for symmetrically enumerating and evaluating possible solutions. Hu [5] solves a related problem where each work element has equal time requirement. In this paper a heuristic method and two methods on finding the shortest route in a network will be presented. The effect of variation of processing times was studied by Mukherjee [10] and recently by Moodie and Young [9]. Held, Karp and Shareshian [3] have given a dynamic programming solution to this problem.

I. PRECEDENCE GRAPHS

Precedence graphs state the precedence relations between work elements due to technological restrictions or restrictions due to fixed facilities or positions of operators. In a precedence graph the work elements are represented by nodes. The nodes are connected by directed arcs showing the precedence relations except that arcs are not drawn where trasitivity implies the precedence. The same relations may be expressed in a matrix form as a precedence matrix of 1, -1 and 0's.

Foulkes [1] discusses the related assembly scheduling problem and uses a mixed graph to represent precedence relations. The basic concept of partitioning the nodes of a graph into a set of equivalence classes is used to enumerate paths. "x and y belong to the same equivalence class if it is possible to trace a directed route from x to y and back again." This definition clearly means that an equivalence class is represented by a directed cycle in the graph. Since it is impossible to have a directed cycle in the precedence graph of a line balancing problem (as none of the jobs around the cycle

could ever be started) and the presence of one indicates inconsistency of precedence relations, the method suggested by Foulkes is irrelevant and wrong as far as assembly line schedules are concerned.

II. A HEURISTIC METHOD

Mukherjee and Basu [11] have modified and simplified Helgeson and Birnie's [4] computerized routine to solve large-scale industrial problems having 100 or more work elements by hand computation. In Phase 1 of this method the work elements are ranked in decreasing order of positional weights. Helgeson and Birnie define the positional weight of a work unit as the sum of its own processing time and of all other work elements which must follow it. Mukherjee and Basu define positional weight of a work element as simply the number of work elements following it. In case of a tie the work unit with a higher processing time is given a higher rank.

In Phase 2, work units are actually assigned to work stations and balance delay is computed from which best balance for each value of n is determined.

$$\text{Balance delay, } d = \frac{nc - \Sigma_i t_i}{nc} \times 100 \text{ per cent}$$

For any fixed n, the computation starts with the ideal value of cycle time. Work elements are assigned in decreasing order of the precedence requirements in the first station as long as it is possible to do so. To fill up the unassigned time a following element is assigned if its predecessors have all been previously assigned. When this is no longer possible assignment starts in 2nd station in the same order. If it is possible to assign all work elements in n stations, the ideal balance is obtained. Otherwise c is increased by 1/100 min. and new computation starts. This is continued until all elements have been assigned to n stations. The solution obtained by this method should not be claimed as optimal but is usually very close to the mathematically optimal solution. In an industrial case study with 94 work elements a balance

delay of 0.75 per cent was obtained. As will be clear later with the ranking by positional weights, Klein's [8] shortest route method could also be used.

III. SHORTEST ROUTE MODEL

Klein [8] has shown how a line balancing problem can be formulated as an assignment problem or as a shortest route problem in a directed network when the order of operations is specified. He suggests that the more general problem can be handled by repeated use of this method when it is possible to enumerate all feasible ordering of operations.

In this method, assuming the given ordering is $1,2,\ldots,n$ a network is constructed with $n + 1$ nodes, with directed arcs (i,j) where $i = 1,\ldots,n$; $i < j$; $j - 2,\ldots,n + 1$ and arc distances $d_{ij} = a_{i,\, j-1}$. If $i \leq j$ and $c - \Sigma_{k=i}^{k=j} t_k \geq 0$, then a_{ij} is the idle time associated with a station to which successive operations i through j have been assigned. In all other cases $a_{ij} = M$. Arcs that are with $a_{ij} = M$ or those with indices $i > j$ are not drawn in the network.

In this network every arc represents a possible assignment of operations to a station and every path from source (1) to sink $(n + 1)$, a possible production line design. Clearly the shortest path from the source to sink gives the solution with min. m delay. The shortest route problem is now solved using the path obtained by starting at the source and going to its closest node; then from this node to its closest, etc., until the sink is reached.

The major drawback of Klein's proposal is the excessive number of shortest routes that must be solved to solve a real-size line balancing problem. Gutjahr and Nemhauser [2] used a similar network to formulate the entire line balancing problem as a single shortest route problem. Also in their network it is sufficient to find any path from the source to sink containing a minimal number of arcs.

They define a 'state' as a partially ordered subset of B which is a collection of work elements that can be processed without prior completion of any other work elements and in any order satisfying p . Let C_i , i = 0, 1,..., r be the entire collection of states with $C_0 = \emptyset$ and $C_r = B$. Each state has a number assigned to it equal to the sum of the processing times of work elements in the state, namely

$$t(C_0) = 0 \quad \text{and} \quad t(C_i) = \Sigma_{x \in C_i} t(x) \qquad i = 1,\ldots, r$$

A directed network N is defined using states (0, 1, 2,..., r) as nodes, with state C_i corresponding to node i . In N there is a directed arc (ij) from nodes i to j if and only if $C_i \subset C_j$ and $t(C_j) - t(C_i) \leq c$ and each directed arc (ij) has a distance $c - (t(C_j) - t(C_i))$.

It is clear that any directed path from 0 to r represents a feasible assignment of all work assignments and the number of arcs the number of stations in the assignment. Also the length of the path represents the total delay time which is to be minimized.

It is proved that for given c the expression [5] is minimized for minimum n . Hence to find a shortest path through N it is sufficient to find any path from 0 to r containing minimal number of arcs. The following procedure gives such a path:

"1. Start at node 0 and construct all arcs (io) from it such that $t(C_i) \leq c$. The nodes reached are called 1st nodes.

2. Suppose the s-th nodes are reached for the first time. For every node among the s-th node, construct an arc to node k if $C_j \subset C_k$ and $t(C_k) - t(C_j) \leq c$. Repeat step 2 with s = s+1 until node r is reached for the first time."

The method is quite elegant for smaller problems but with an increasing number of work elements and flexibility of precedence relations the 'generation of all states' itself becomes an enumerative problem and computation on time

increases very rapidly.

IV. CONCLUSION

There is a close relation between the above method [2] and the dynamic programming approaches of Jackson [6] and Held et al [3]. However as with usual dynamic programming problems the computational difficulties multiply as the number of states increases. It is evident that any method for line balancing which is basically enumerative will face this problem of computational hurdles and these methods may not be economical to use for a moderately large problem, though the method of Held et al is quite powerful. It is seen that heuristic approaches as of Helgeson and Birnie or Mukherjee and Basu still possess certain advantages over their analytical counterparts. They are simpler, less time-consuming and give a resultant balance-delay which is small enough for any practical purpose. Also sensitivity analysis is simplified. The real assembly line balancing problem is further involved due to (a) variation of processing times from cycle to cycle and from operator to operator, (b) error in measuring or estimating processing times of elements, and (c) non-additivity of processing times for work elements (i.e., $t_a + t_b + t_c \neq t_a + t_c + t_b$ if performed in that order). Hence further refinement of balance at increased cost is often undesirable. It seems that effort should be directed to include these other features of the line balancing problem in a model. Such a model will be much more realistic so that we can definitely say that the balance achieved on paper will be realized in practice which is often not true for the models available now.

REFERENCES

[1] Foulkes, I.D., "Directed Graphs and Assembly Schedules," Proc. Symp. in Appld. Math., Vol X, 281-289, (1960).

[2] Gutjahr, A.L. and Nemhauser, G.L., "An Algorithm for the Line Balancing Problem," Mgt. Sci., 11 (2), 308- 315, (1964).

[3] Held, M. et al, "Assembly Line Balancing - Dynamic Programming with Precedence Constraints," Opns. Res., 11 (3), 442-459, (1963).

[4] Helgeson, W.B. and Birnie, D.P., "Assembly-Line Balancing Using Ranked Positional Weight Technique," J. Industr. Engr., 12 (6), 394--, (1961).

[5] Hu, T.C., "Parallel Sequencing and Assembly Line Problems," Opns. Res., 9 (6), 841-48, (1961).

[6] Jackson, J.R., "A Computing Procedure for a Line-Balancing Problem," Mgt. Sci., 2 (3), (1956).

[7] Kilbridge, M.D., and Wester, L., "A Review of Analytical Systems of Line Balancing," Opns. Res., 10, 626-638, (1962).

[8] Klein, M., "On Assembly Line Balancing," Opns. Res., 11 (2), 276-281, (1963).

[9] Moodie, C.L., and Young, H.H., "A Heuristic Method of Assembly Line Balancing for Assumptions of Constant or Variable Work Element Times," J. Indstr. Engr., 16 (1), 23-29, (1965).

[10] Mukherjee, S.K., "A Heuristic Approach of Assembly-Line Balancing," M. Tech Thesis, Indian Institute of Technology, Kharagpur, India, July 1962.

[11] Mukherjee, S.K., and Basu, S.K., "An Application of Heuristic Method of Assembly Line Balancing in an Indian Industry," Proc. Inst. of Mech. Engrs., London, P 11/64, (1964).

[12] Salveson, M.E., "The Assembly-Line Balancing Problem," Trans. Amer. Soc., Mech. Engrs., 77, 939, (1955).

SUMMARY OF KLEE'S RESULTS ON BOUNDING THE NUMBER OF SIMPLEX CYCLES

by Bennett Fox

A linear program is nondegenerate if the b vector cannot be expressed as a linear combination of fewer than m columns of the $m \times n$ constraint matrix which implies that no $\bar{b}_j$ ever vanishes. We assume throughout that we start with a feasible vertex, although by reinterpretation of "feasible" and "optimal," the results apply to Phase 1 as well as to Phase 2. Let $LP(m,n)$ be the class of nondegenerate linear programs with m linear equality constraints and n $(0 < m < n)$ non-negative variables (including slacks).

Using a coordinate-free, geometric approach, Victor Klee [2] shows that there exists a linear program in $LP(m,n)$ and a feasible starting vertex such that $m(n - m - 1) + 1$ simplex iterations are required to reach the optimal vertex when the widely used steepest descent rule determines the column to enter the basis.

No numerical examples are given in [2], but Phillip Wolfe [4] has given the one below for the case $m = 1$:

$$\text{Max} \quad \sum_{j=1}^{n-1} 2^{-j} x_j$$

$$\text{s.t.} \quad \sum_{j=1}^{n-1} 4^{-j} x_j + x_n = 1$$

$$x_j \geq 0 , \quad j = 1, \ldots, n .$$

The "min $\bar{c}_j$" rule (steepest descent) pivots on $x_1, x_2, \ldots, x_{n-1}$ in succession. Thus, with one constraint, we can go through every possible basis.

Klee conjectures that $m(n - m - 1) + 1$ is an upper bound (necessarily sharp when using steepest descent) on the number of simplex iterations to reach the optimal vertex for any program in $LP(m,n)$ with any rule for introduction of new basic variables that decreases the value of the objective function at each stage (i.e., the pivot columns always have $\bar{c}_j < 0$). The conjecture has been proved [3] only for $n < m + 4$. This bound (if correct) is a great improvement over the trivial upper bound $\binom{n}{m}$.

Comments:

1. The proof of the conjecture for $n < m + 4$ seems to work only when the feasible region is bounded. However to guarantee that this holds it suffices to add the condition $\sum_{j=1}^{n} x_i + x_{n+1} = k < \infty$. Here x_{n+1} is a nonnegative slack variable and k is chosen large enough, say 10^{30}, such that if the constraint ever becomes tight we will assume that the objective function was unbounded from below in the original feasible region. The conjectured upper bound on the number of simplex cycles is now $(m + 1)(n - m - 1) + 1$.
2. By perturbation we can always ensure nondegeneracy. However, to save time all existing codes omit this and hence with these codes the sharp upper bound on the number of simplex cycles is ∞, although this apparently occurs only for specially constructed pathological examples. In practice, the number of simplex cycles rarely exceeds $3m$. See [1].

Note that Klee's conjecture applies *only* to nondegenerate programs.

REFERENCES

[1] Dantzig, G.B., *Linear Programming and Extensions*, Princeton University Press, Princeton, N.J., 1963.

[2] Klee, V., "A Class of Linear Programming Problems Requiring a Large Number of Iterations," Boeing Scientific Research Labs., Report D1-82-0387, 1964.

[3] ----, "Heights of convex polytopes," J. Math. Anal. Appl., 10, 1965.

[4] Personal communication to G.B. Dantzig.

REVIEW:

CYCLING IN THE TRANSPORTATION PROBLEM

By B. J. Gassner, *Naval Research Logistics Quarterly*, 11, 43-58, (1964).

This paper contains a numerical example of cycling (circling) in a 4×4 assignment problem. However, to construct the class of problems from which this example was taken required the use of rather artificial rules for adding and dropping basic variables. For example, in going from cycle 2 to cycle 3 x_{42} entered, but $\bar{c}_{32} = -4$ whereas $\bar{c}_{42} = -2$. Thus, the steepest descent criterion was not used.

The rule for choosing the variable to drop in case of ties was also unnatural. The choice must be such that, fixing at its current value the price on the row where the new basic variable is to enter, only nonnegative (nonpositive) quantities are to be added to the current row (column) prices.

The 3×3 assignment problem, under a very special hypothesis, is shown to be incapable of circling when the drop rule described above is used. This special hypothesis is that the value of $\bar{c}_{ij}^{\,k}$, where $x_{ij}^{\,k}$ is to enter the basis on the $(k + 1)$st cycle, is the same for all k. Since this condition is so peculiar, the problem should be considered open.

— Bennett Fox

RESEARCH PROBLEMS

by George B. Dantzig

The following is a list of eight well-known unsolved problems from the field of mathematical programming. The original list compiled in August 1962, contained ten problems, two of which have since been solved.

1. Let C_n be an n-dimensional bounded polyhedral convex set defined by 2n distinct faces n of which determine the extreme point p_1 and the remaining n of which determine the extreme point p_2. Does there always exist a chain of edges joining p_1 to p_2 such that the number of edges in the chain is n ?

2. Let E be the extreme points of a unit n-cube having as faces the coordinate hyperplanes through the origin and the hyperplanes parallel to them passing through the point (1, 1,..., 1). Let P be a given hyperplane which separates E into two parts, E_1 and E_2. Characterize the n-1 dimensional faces of the convex set having E_1 as its complete set of extreme points.

3. A matrix such that the determinant of every square submatrix has value -1, 0, or +1 is called *unimodular*. The distinct columns of such a matrix are said to form a *complete set* if the annexation of a column not in the set destroys the unimodular property. Two such sets *belong to the same class* if one can be obtained from the other by a permutation of the rows of its matrix. Characterize the various classes. How can they be generated? Given a matrix, find necessary and sufficient condition that it satisfy the unimodular property.

4. Given an $n \times n$ permutation matrix $[x_{ij}]$; i.e., a zero-one matrix each

row and column of which have exactly one unit element. Let the set of its n^2 elements constitute a point in n^2-dimensional coordinate space. It is known that these and only these points are extreme points of the convex set defined by

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad j = 1,2,\ldots,n$$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad i = 1,2,\ldots,n$$

$$x_{ij} \geq 0$$

It is not known, however, how to characterize the faces of the convex polyhedral sets whose extreme points consist only of those permutation matrices which represent n-cycles.

5. If a square matrix M has k nonzero elements, k multiplications and additions are required to multiply M by an arbitrary vector b. Its inverse M^{-1}, which is assumed to exist, may have more than k nonzero elements, hence may require more than k of each of these operations if multiplied by an arbitrary vector b. However, M^{-1} may be represented (in fact, in many ways) as a product of elementary matrices generated by row and column transformations on M. Prove or disprove the conjecture that there always exists a representation such that carrying out the indicated operations with b will require no more than $\ell = k$ multiplications and $\ell' = k$ additions. If not true, find sharp lower bounds for ℓ and ℓ'.

6. Given an n-dimensional simplex with n-1 of its n faces fixed and its remaining face variable but parallel to a given hyperplane. Find the

largest such simplex whose interior contains no lattice point.

7. Geometrically, the simplex method moves from one vertex to that of a neighboring vertex which gives the greatest change in the value of the linear form (where variables correspond to the coordinates of a point). For convex sets defined by m equations in n non-negative variables, the number of such moves is remarkably low (often between m and n in practice). Intuitively, wandering on the outside of a convex set in this manner would appear to be extremely inefficient, yet empirical evidence from thousands of cases is to the contrary. Why?

8. Usually, there is an analogue for linear inequality systems of a theorem about linear equality systems. The problem of minimizing a positive definite quadratic form subject to linear equation constraints is reducible (by the method of Lagrange) to solving a certain system of linear equations. If the constraints are replaced by linear inequalities, is the entire problem reducible to that of solving a certain system of linear inequalities (i.e., to an equivalent linear program)?

Following are some abstracts of dissertations and theses submitted to the University of California at Berkeley on subjects pertaining to Operations Research.

Multi-Stage Programming under Uncertainty

By

Mostafa Mohamed Nabih El-Agizy

ABSTRACT

This investigation is concerned with a multi-stage linear programming problem under uncertainty. The model considered is an extension to the two-stage problem first formulated by Dantzig. In each period demand is a random variable with known distribution function. The problem is characterized by decisions made sequentially over a horizon of k time periods. In each period to make the best possible decision one must use the observed values of the random parameters. When demand has a discrete distribution we have established an approach that reduces the multi-stage problem to a linear program with upper bounded variables. This results in a meaningful reduction in problem size. Even so the latter program can grow very large, so that an additional algorithm, using the decomposition principle, is proposed as a solution technique.

Several special cases of practical importance that fit in this formulation are considered:

a- A transportation model

b- A production-inventory model

c- A Leontief substitution model of production

d- A dynamic inventory model.

For the two-stage transportation model it is shown that the corresponding

deterministic equivalent program represents a directed network. For the two-stage production-inventory problem we show that the equivalent linear program is particularly simple. Moreover, for the Leontief production model, it is shown that the optimal selection of alternative substitute activities is independent of the demand distribution. However, the above results do not seem to extend to the multistage case. Finally, we establish that the dynamic inventory problem has a network representation; thus network flow theory can be used for solving this class of problems.

An Adaptive Age Replacement Policy

By

Bennett Fox

ABSTRACT

Under an age replacement policy we replace at failure or at the end of a specified time interval, whichever occurs first. This makes sense if a failure replacement costs more than a planned replacement and the failure rate is strictly increasing. We assume that the failure distribution is a Weibull distribution with known shape parameter (greater than one) and unknown scale parameter. Moreover, we assume that a natural conjugate prior distribution with specified parameters is at hand which we modify after each stage according to Bayes' rule. Our policy adapts to the changing prior. We see intuitively that the larger the replacement interval set, the more information we are likely to obtain. We take account of this in a precisely defined manner via dynamic programming. The optimal policy is partially characterized and various limiting results are obtained.

Games of Sequence Prediction

By

David William Matula

ABSTRACT

A predictor is allowed to observe a sequence indefinitely and must pick some point at which he believes a particular subsequence will not occur next. In order to insure a guaranteed reliability the predictor assumes the sequence is generated by an emittor who is trying to have this subsequence occur.

We show that if the subsequence involved has a "self disjoint" property, i.e. it does not overlap with itself, and is composed of $K \geq 2$ characters, $N \geq 2$ of which are distinct, then the value of this sequence prediction game is $(K-1)^{K-1}/K^K$ for any N ; the emittor has an encoding strategy that assures passage of the subsequence with probability at least $(K-1)^{K-1}/K^K$ against any prediction strategy; and the predictor has a strategy which limits the probability of occurrence of the subsequence to $(K-1)^{K-1}/K^K+a$ for any $a>0$ no matter what sequence is presented. It is observed that for long "self disjoint" subsequences the optimal encoding achieves a reliability of only $1/e$ times the return which could be achieved against a random prediction strategy.

Even in some cases where the subsequence the emittor is trying to pass has a relatively simple structure, we show that the emittor's optimal encoding strategy can require an "infinite memory" of previously emitted characters. The predictor's a-optimal strategies require only a finite memory which, however, grows as a approaches zero.

Wildland Fire Control with Limited Suppression Forces

By

Alan Wayne McMasters

ABSTRACT

The studies of Parks and Jewell are extended to the problems of initial attack allocation of limited suppression forces, including the development of risk models for two-stage programming under uncertainty for several realistic allocation strategies using event-review schedules. The strategies allowing transfer of forces between fires are shown to be better than those allowing no transfers.

The analyses begin with the development of a deterministic model for two classes of strategies for allocating forces to a new fire; those where reserves are the only source of forces, and those where the source includes both reserves and forces assigned to existing fires. Allocation policies are then derived which minimize the total cost of all fires for the special case of two non-simultaneous fires. These policies are used in the development of two-stage risk models which is then used to determine the optimal initial attack force to assign to an existing fire under the threat of additional fires.

The salient features of the risk models are illustrated in a numerical example in which parametric variation of several parameters allows the explanation of a large region of possible behavior for the strategies. The size of the suppression force to send to a current fire is shown to decrease as the inter-fire transportation cost increases. The general strategies allowing transfer

of forces between fires are shown to be better than those allowing no transfer of forces, and, as is expected, the total expected costs are shown to increase as the size of the available suppression forces decrease.

Stochastic Wear Processes

By

Richard C. Morey

ABSTRACT

A new class of non-decreasing stochastic processes is characterized. These processes satisfy a generalization of the notion of an increasing failure rate. From physical considerations, these processes seem suitable for describing the process of cumulative wear or damage. The main interest with the model is an investigation of the first time until the process exceeds a random barrier.

For this class of processes, it is shown that the first passage time random variable across a random barrier has an increasing failure rate, regardless of the distribution of the barrier. In addition, by the use of certain intuitive, non-parametric assumptions, tight bounds on the moments of this first passage time random variable are obtained.

Bounds for Lattice Distributions Having Monotone Hazard Rate with Applications

By

Gamanlal P. Shah

University of California, Berkeley

ABSTRACT

Using methods which have been employed for obtaining bounds on continuous failure distributions, bounds on lattice distributions are derived under the increasing (decreasing) failure rate assumptions. The discrete bounds are convenient in a number of applications such as life table analysis and reliability theory.

Mathematical Programming and Optimal Control

By

Richard Maurice Van Slyke

ABSTRACT

Let K be a closed conved set in E^{m+1} and $L = \{P = (P_0,\ldots,P_m) : P_1 = P_2 = \ldots = P_m = 0\}$. Then for the simple problem:

$$\text{minimize } P_0$$
$$\text{subject to } P = (P_0,P_1,\ldots,P_m) \in K \cap L ,$$

we prove a duality theorem and the convergence of a solution algorithm modeled on the duality theorem and the simplex method of linear programming respectively.

Specialization of this general model to linear programming, convex programming, generalized programming, control theory, and the decomposition approach to mathematical programming yield the appropriate duality theorems and solution algorithms in each case.

The principle idea exploited here is the notion of supporting hyperplanes to convex sets. The duality theorem is a direct application of the fact that every boundary point of a convex set belongs to a supporting hyperplane; moreover, the generalized simplex method presented here is most useful when K is characterized in such a way that, given a hyperplane, the translate of it which is a supporting hyperplane of K may easily be found, if it exists, as well as the points of the supporting hyperplane which

are common to the boundary of K. This corresponds to "pricing out" in linear programming.

We show that many problems in control theory are special cases of our model and for a large class of linear control problems a solution method is outlined to illustrate the use of mathematical programming techniques to solve optimal control problems.

Two appendices contain the elements of the theory of affine spaces, of convex sets, and of ordinary differential equations used in the text.

An Inventory Model for Rented Equipment

By

William Donald Whisler

ABSTRACT

An inventory of rented equipment is studied. Equipment is withdrawn from the inventory by customers who use it for a length of time and then return it. Decisions about the amount of equipment to rent can be made at certain points in time. This dissertation describes a policy for making these decisions which minimizes expected costs.

A dynamic programming model is formulated to describe the problem. The model is different than the usual ones considered in the literature in three respects. First, the equipment which is withdrawn from the inventory is not consumed; it is only used for a certain length of time then it returns to the inventory. The amount of equipment in the inventory, consequently, can fluctuate *up or down*. Second, all the equipment in the inventory is rented. Thus, when a decision is made about how much equipment is needed, either *more or less* than currently is on hand can be rented. Third, convexity of the cost function is not important because simple optimal policies can be found when the cost function is nonconvex.

The exact form of the optimal policy depends on the specific assumptions made, however, all of the optimal policies have the following generic structure. At the time a decision is to be made two numbers t and u, with $t \leq u$, can be computed. If the amount of equipment currently on hand is

greater than u (less than t) then it is optimal to rent $u(t)$. If the amount of equipment is in between t and u then, assuming convexity, it is optimal to continue renting the same quantity or, assuming nonconvexity, no general statement can be made about what to do because for each number rented in between t and u there may be a different optimal number to rent.

Sensitivity Analysis of Networks

By

Richard D. Wollmer

ABSTRACT

This investigation is concerned with a sensitivity analysis on source-sink planar maximum flow networks. Specifically each arc of the network is subject to breakdowns which result in a reduction in its capacity. The problem is to find the greatest reduction in maximum flow possible if n breakdowns occur and where they must occur for this to happen.

The network itself is defined by a set of arcs and a set of points called nodes. Each arc joins two nodes and has associated with it a positive capacity which represents the maximum amount of flow that may pass over it. One of the nodes is designated as the source and another as the sink. These nodes, arcs, and capacities are sufficient to determine the maximum amount of flow that may pass from source to sink. In addition, it is required that if an arc joining the source to the sink is added, the new network may be drawn on a sphere in such a way that no two arcs intersect except at a node.

Two cases are considered for the sensitivity analysis problem. In the first case the problem is solved exactly where it is assumed that each arc is subject to one breakdown and the amount by which the capacity of an arc is reduced due to a breakdown is a deterministic quantity.

In the second case the problem is solved approximately where multiple breakdowns are allowed for the individual arcs but the amount by which the

capacity of an arc is reduced due to breakdowns is a random variable with unknown distribution but with known mean and variance.

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* University of California, Berkeley	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

NOTES ON OPERATIONS RESEARCH -- 5

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Research Report

5. AUTHOR(S) *(Last name, first name, initial)*

Various

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
April 1966	53	25

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
Nonr-222(83) b. PROJECT NO. NR 047 033 c. RR 033-02-01 d.	ORC 66-13 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. AVAILABILITY/LIMITATION NOTICES

Available without limitation on dissemination.

11. SUPPLEMENTARY NOTES	12 SPONSORING MILITARY ACTIVITY
	Mathematical Science Division

13. ABSTRACT

See pages 36-49.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Random sampling						
Dual programs						
Assembly line						
Complementary solutions						
Bounding the number						

INSTRUCTIONS

1. **ORIGINATING ACTIVITY:** Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. **REPORT SECURITY CLASSIFICATION:** Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. **GROUP:** Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. **REPORT TITLE:** Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. **DESCRIPTIVE NOTES:** If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. **AUTHOR(S):** Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. **REPORT DATE:** Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. **TOTAL NUMBER OF PAGES:** The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. **NUMBER OF REFERENCES:** Enter the total number of references cited in the report.

8a. **CONTRACT OR GRANT NUMBER:** If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. **PROJECT NUMBER:** Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. **ORIGINATOR'S REPORT NUMBER(S):** Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. **OTHER REPORT NUMBER(S):** If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. **AVAILABILITY/LIMITATION NOTICES:** Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. **SUPPLEMENTARY NOTES:** Use for additional explanatory notes.

12. **SPONSORING MILITARY ACTIVITY:** Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. **ABSTRACT:** Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. **KEY WORDS:** Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.